

10037251.022702

1 **CLAIMS**

2 1. In a computer device that uses flash memory to store data, a method
3 comprising:

4 maintaining one or more mapping data structures containing mappings of
5 logical flash memory addresses to physical flash memory addresses, each mapping
6 data structure having a predetermined capacity of mappings;

7 maintaining a master data structure containing a pointer to each of the one
8 or more mapping data structures;

9 allocating additional mapping data structures as needed to provide capacity
10 for additional mappings.

11
12 2. The method as recited in Claim 1, further comprising adding pointers to the
13 master data structure for the additionally allocated mapping data structures.

14
15 3. The method as recited in Claim 1, wherein the mapping data structures and
16 master data structures are generated by a flash driver.

17
18 4. The method as recited in Claim 1, wherein the mapping data structures and
19 master data structures are stored in a volatile memory device of the computer.

20
21 5. The method as recited in Claim 1, further comprising removing one or
22 more additionally allocated mapping data structures if the capacity of mappings is
23 not needed.

24
25 6. A system for tracking data in a flash medium, comprising:

1 a secondary data structure containing logical sector address to physical
2 sector address mappings showing a relationship between logical sector addresses,
3 requested by a file system, to physical sector addresses in which associated data is
4 physically stored on the flash medium;

5 a master data structure containing at least one pointer that points to at least
6 one secondary data structure; and

7 means for allocating a third data structure, if the secondary data structure
8 becomes full, wherein the third data structure contains logical sector address to
9 physical sector address mappings.

10
11 7. The system as recited in Claim 5, wherein the master data structure is
12 configurable to contain a pointer that points to the third data structure, if allocated.

13
14 8. The system as recited in Claim 5, further comprising a flash media driver
15 configured to determine how many physical sectors are contained on the flash
16 medium.

17
18 9. The system as recited in Claim 5, wherein the means for allocating the third
19 data structure is a flash driver configured to monitor how many logical sector
20 address requests are issued by the file system to ensure there is enough data
21 structure(s) allocated in addition to the secondary data structure.

22
23 10. The system as recited in Claim 1, wherein the means for allocating the third
24 data structure is also configured to remove the third data structure in the event the
25

1 secondary data structure is sufficient for mapping physical sector addresses
2 containing data to logical sector addresses.

3
4 11. The system as recited in Claim 1, further comprising means for allocating a
5 fourth data structure, if the second and third tables are full.

6
7 12. The system as recited in Claim 1, wherein the tables are stored in a volatile
8 memory device.

9
10 13. A system, comprising:

11 a master data structure containing 1 to N pointers, wherein N is an integer
12 greater than 1; and

13 a secondary data structure containing mappings of logical sector addresses
14 to physical sector addresses, the logical sector addresses contained in the
15 secondary data structure being a portion of the maximum possible quantity of
16 logical sector addresses that can be issued by the file system, wherein at least one
17 of the pointers in the master data structure points to the secondary data structure.

18
19 14. The system as recited in Claim 13, further comprising a third data structure
20 containing mappings of logical sector addresses to physical sector addresses,
21 allocated by the flash driver when the portion of logical sector addresses contained
22 in the secondary data structure is insufficient to store logical sector address write
23 requests issued by the file system.

1 15. The system as recited in Claim 13, further comprising a third data structure
2 containing mappings of logical sector addresses to physical sector addresses,
3 allocated by the flash driver when the portion of logical sector addresses contained
4 in the secondary data structure is insufficient to store logical sector address write
5 requests issued by the file system, wherein at least one the pointers in the master
6 data structure points to the third data structure.

7
8 16. The system as recited in Claim 13, further comprising a third data structure
9 containing mappings of logical sector addresses to physical sector addresses,
10 allocated by the flash driver when the portion of logical sector addresses contained
11 in the secondary data structure is insufficient to store logical sector address write
12 requests issued by the file system and de-allocated by the flash driver if the portion
13 of logical sector addresses contained in the secondary data structure becomes
14 sufficient to store the logical sector address write requests issued by the file
15 system.

16
17 17. The system as recited in Claim 13, wherein the flash driver comprises a
18 flash abstraction layer configured to monitor logical sector address requests by the
19 file system and update the mappings of logical sector addresses to physical sector
20 addresses.

21
22 18. The system as recited in Claim 13, wherein the master and secondary tables
23 are stored in a volatile memory device.
24
25

1 19. The system as recited in Claim 13, wherein the master and secondary tables
2 are stored in a random access device.

3
4 20. The system as recited in Claim 13, wherein the flash driver comprises a
5 flash media layer, configured to determine a size of the flash medium.

6
7 21. A computer device, comprising:

8 a flash driver configured to serve as an interface between a file system and
9 the flash memory medium;

10 a master data structure containing enough pointers to match a maximum
11 quantity of logical sector addresses to be issued by the file system; and

12 a secondary data structure containing mappings of logical sector addresses
13 to physical sector addresses, the logical sector addresses contained in the
14 secondary data structure being a portion of the maximum possible quantity of
15 logical sector addresses to be issued by the file system, wherein each of the
16 pointers from the master data structure points to a specific mappings of logical
17 sector address to physical sector addresses.

18
19 22. The computer device as recited in Claim 21, further comprising a third data
20 structure containing mappings of logical sector addresses to physical sector
21 addresses, allocated by the flash driver when the portion of logical sector
22 addresses contained in the secondary data structure is insufficient to store logical
23 sector address write requests issued by the file system.

1 23. The computer device as recited in Claim 21, further comprising a third data
2 structure containing mappings of logical sector addresses to physical sector
3 addresses, allocated by the flash driver when the portion of logical sector
4 addresses contained in the secondary data structure is insufficient to store logical
5 sector address write requests issued by the file system, wherein at least one the
6 pointers in the master data structure points to the third data structure.

7
8 24. The system as recited in Claim 21, further comprising a third data structure
9 containing mappings of logical sector addresses to physical sector addresses,
10 allocated by the flash driver when the portion of logical sector addresses contained
11 in the secondary data structure is insufficient to store logical sector address write
12 requests issued by the file system and de-allocated by the flash driver if the portion
13 of logical sector addresses contained in the secondary data structure becomes
14 sufficient to store the logical sector address write requests issued by the file
15 system.

16
17 25. The computer device as recited in Claim 21, wherein the flash driver
18 comprises a flash abstraction layer configured to monitor logical sector address
19 requests by the file system and update the mappings of logical sector addresses to
20 physical sector addresses.

21
22 26. The computer device as recited in Claim 21, wherein the master and
23 secondary tables are stored in a volatile memory device.
24
25

1 27. The computer device as recited in Claim 21, wherein the master and
2 secondary tables are stored in a random access device.

3
4 28. The computer device as recited in Claim 21 wherein the computer device is
5 portable data storage and processing device.

6
7 29. In a computer device that uses flash memory to store data, a method
8 comprising:

9 generating a master data structure containing a plurality of pointers;

10 allocating a secondary data structure used to map logical sector addresses to
11 physical sector addresses, wherein the secondary data structure is limited in size;

12 enabling one of the plurality of pointers to point to the secondary data
13 structure;

14 allocating a third data structure used to map logical sector addresses to
15 physical sector addresses, if the secondary data structure fills-up; and

16 enabling one of the plurality of pointers point to the third data structure, if
17 allocated.

18
19 30. The method as recited in Claim 29, wherein the logical sector addresses are
20 issued by a file system and the physical sector addresses indicate where data
21 associated with the logical sector addresses is physically stored on the flash
22 medium.

1 31. The method as recited in Claim 29, further comprising ascertaining a
2 quantity of physical sectors on the flash medium prior to generating the secondary
3 data structure.

4
5 32. The method as recited in Claim 29, further comprising ascertaining a
6 quantity of physical sectors on the flash medium prior to generating the secondary
7 data structure and determining an address bit length for the pointers in relation to
8 the quantity of physical sectors ascertained.

9
10 33. The method as recited in Claim 29, wherein the secondary data structure is
11 $b*k$ bytes in size, wherein k is a number of physical sector addresses contained in
12 the data structure and b is a number of bytes required to store each physical sector
13 address.

14
15 34. One or more computer-readable media comprising computer-executable
16 instructions that, when executed, perform the method as recited in Claim 29.

17
18 35. The method as recited in Claim 29, wherein the computer device is portable
19 processing device.

20
21 36. The method as recited in Claim 29, wherein the method is performed by a
22 flash driver in conjunction with the file system of the computer device.

23
24 37. The method as recited in Claim 29, wherein the tables are stored in a
25 volatile memory portion of the computer device.

1
2 38. The method as recited in Claim 29, wherein the secondary data structure
3 fills-up when the logical sector addresses exceed the limited size of the secondary
4 data structure.

5
6 39. A computer-readable medium for a Flash driver, comprising computer-
7 executable instructions that, when executed, direct the Flash driver to:

8 generate a master data structure containing a plurality of pointers;

9 allocate a secondary data structure used to map logical sector addresses to
10 physical sector addresses, wherein the logical sector addresses are issued by a file
11 system and the physical sector addresses indicate where data associated with the
12 logical sector addresses is physically stored on the flash medium;

13 enable one of the plurality of pointers point to the secondary data structure;

14 allocate a third data structure used to map logical sector addresses to
15 physical sector addresses, if the secondary data structure fills-up; and

16 enable one of the plurality of pointers point to the third data structure, if
17 allocated.

18
19 40. The computer-readable medium as recited in Claim 39, further comprising
20 computer-executable instructions that, when executed, direct the Flash driver to
21 allocate one or more additional data structures in the event that the third data
22 structure fills-up.